

Abstract

The invention provides an efficient method and apparatus for exchanging data between two entities, such as two application programs or a resource library and an application program. The invention provides a mechanism for populating a complex object model implementation in accordance with the particular data format preferred by the receiving entity from one or more external data formats on the fly responsive to a request for data. The method and apparatus allow the registration of resource factories which can produce resources through mechanisms other than the traditional parsing of a file. One implementation of this is using the URL scheme to identify the factory to handle a request, and allowing that factory to create a specialized Java package resource.

The method and apparatus allow the registration of reflection adapter factories, and specialized model classes which work with those adapters, for the purpose of computing object attributes and populating the Java package resource with them. One implementation of this mechanism is a set of Java Reflection adapters which compute the properties of JavaClass model objects from the JDK reflection API's.